U.S. DEPARTMENT OF COMMERCE
National Institute of Standards a-- T--

AD-A261 195

NISTIR 4684

National PDES Testbed Report Series

A Proposed Testing Methodology for STEP Application Protocol Validation

NATIONAL PDES TESTBED

93-04000

NIST

93 2 25

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards and Technology

NISTIR 4684

National PDES Testbed

NATIONAL PDES TESTBED

A Proposed Testing Methodology for STEP Application Protocol Validation

Mary Mitchell

U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary of Commerce

National Institute of Standards and Technology
John W. Lyons, Director

Sept 26, 1991

Table of Contents

List of Figures

List of Tables

Abstract

An Application Protocol (AP) is a specification for a subset of the data described by the Standard for the Exchange of Product Model Data (STEP). Application Protocols are designed to permit practical implementations of STEP. Validation is needed to ensure that the technical solutions provided by the AP will work in a practical sense. This document proposes that the STEP development policy be strengthened to require that Application Protocols be validated prior to being submitted for standardization. Justification for this additional requirement on Application Protocols is provided. The body of the paper describes a series of validation techniques that are appropriate for the development methods used by STEP. A process is proposed under which these validation techniques should be applied. In addition, this paper describes the contribution that AP validation could make to conformance testing.

A Proposed Testing Methodology for STEP Application Protocol Validation

Mary Mitchell

I. Introduction

Confidence in a standard by its user community is absolutely essential for a standard to gain acceptance. Proof that the standard is properly defined and that it can be used successfully will help achieve user confidence. A rigorous testing program is the foundation for any useful standard. Appropriate testing before standardization can ensure that a draft specification indeed meets the functional requirements for the standard. This type of testing will necessarily be distinct from the testing of implementations for conformance to the standard.

The Standard for the Exchange of Product Model Data (STEP)[1] is an emerging international standard designed to provide a complete, unambiguous, computer-readable definition of the physical and functional characteristics of a product throughout its life cycle. STEP model specifications are implementation independent, though distinct implementation interface techniques are defined to support applications based on file exchange or shared databases.

An Application Protocol (AP) is a specification for a subset of the data described by STEP. This subset of STEP entities and the corresponding usage constraints describe the product data requirements of a given application [WG4-N15]. Thus the STEP AP specifications permit product information to be unambiguously exchanged or shared between implementations on dissimilar systems.

As STEP is being developed, procedures are in place to ensure that the AP specifications are quality documents. But procedures are not in place to ensure that the technical solution that an AP provides will work in a practical sense.

This paper will show that requiring the validation testing of each STEP AP during its development is a cost effective way to ensure that STEP is free of technical flaws <u>before</u> the specifications become standards. For this requirement to be placed on STEP AP developers, a practical methodology for executing the validation testing must be available. The methodology proposed in this paper builds on established software testing techniques, based on previous experience with testing of STEP subsets, specifically developed to be compatible with the methods used in developing STEP. In addition, this paper describes how the outputs of the validation of an AP

[1] STEP is under development by the International Organization for Standardization (ISO) Technical Committee 184 (TC184)/Sub-Committee 4 (SC4).

can contribute to the future conformance testing of an implementation based on the AP.

This paper is directed at three audiences:

- developers of Application Protocols and AP methods,
- contributors to the testing projects within ISO TC184/SC4 and the IGES/PDES Organization[2] and
- developers of software that could contribute tools for validation testing.

Section II discusses the role of validation testing including how validation testing has been used in software development and how it can be applied in the development of STEP. A detailed discussion of the proposed testing methodology is provided in Section III. Section IV describes how validation relates to STEP conformance testing. Some concepts and some terminology are still evolving as STEP is developed. The terms used in this report are defined in Section VI. Brief descriptions of the foundation work that has been done in the STEP community are given in the appendices.

II. The Role of Validation Testing

This section briefly defines what is meant by the validation testing of STEP. How validation testing can best be accomplished in the development of STEP is still being debated in the STEP community. The rationale for validation testing is summarized along with the results of a literature search for techniques that could be utilized in testing the usefulness of STEP. The basis for the proposed methodology, including a discussion of when it is most cost effective to test STEP, is provided.

The Need for Validation Testing

Imagine two contractors that need to exchange product design data. They are working on a contract that requires the exchange of product data using STEP. Unfortunately these exchanges result in an insufficient transfer of data. The cause is determined to be a defect in STEP. In order to fulfill the requirements of their contract, they must negotiate a change to the standard. Eventually this local change may work its way back into the official standard. More often, each application system is required to tailor translation software for every other application system with which it is expected to exchange data. In fact, examples such as this are more than mere speculation. Defects in the Initial Graphics Exchange Specification (IGES) [NBS] resulted in the "flavoring", e.g., the tailoring of translation software, of IGES [CAL1,

[2] The IGES/PDES Organization (IPO) is a U.S.- based standards activity that has made significant contributions to the development of STEP.

Jur89]. In IGES, the predecessor to STEP, a number of limitations were not uncovered until vendors tried to implement the standard.

Significant investment will be required to implement STEP-based applications. In addition, STEP may contain a number of untried solutions to technical problems. The STEP user community must not ask vendors to develop implementations based on specifications that have unknown levels of risk. Thorough, appropriate testing before standardization can minimize the need to continually 'patch' the standard to correct design flaws uncovered by implementing the standard.

STEP needs to be free of major defects. If we wait until STEP is a standard to prove robustness, the possibility that STEP will fail is greatly increased. Confidence that the standard is properly defined and that it can be used successfully is required before the standard will be accepted and products built to its specifications. Such proof of feasibility will accelerate the acceptance and implementation of STEP.

A rigorous testing program is the foundation for any useful standard. Techniques are needed to properly test the specifications against the initial requirements for STEP - before the specifications become standards. This type of testing, validation testing, is necessary in addition to the more well known conformance testing.

Thus, the validation testing of STEP is a process designed to determine whether an application protocol does what it is intended to do, i.e., meets the requirements that led to its development. If it is considered a part of the actual development of the application protocol it can more effectively contribute to the robustness of STEP.

The Lack of Existing Techniques

The fundamental technology used for developing STEP is information modeling. Information modeling uses a set of formal techniques [ISO11, ICA82, ICA85, Nij89, WG5-N26] for describing information requirements. The STEP development process uses these techniques to achieve consensus and to document the results of that consensus. The resulting specifications are written in the language Express [ISO11]. These specifications are independent of implementation detail (i.e., they support both file exchange and shared database implementations) and could be thought of as a detailed design for data exchange. "STEP should be standardized only when the information models in it are proven to be robust enough to support a minimum range of application uses in key product life cycle scenarios." [Hen89]

Techniques exist for software testing that are designed to evaluate implementations[3] of specifications which were developed with a particular implementation technology and strategy in mind. Work also exists on verification[4] and validation of knowledge acquisition systems [Bah91]. However, techniques for validation testing of implementation-independent information models could not be found in the literature by this author. Some verification techniques do exist [Nij90, Jor91] but only reports from early STEP testing activities (see Appendix B) describe a process for validating the usability of an information model [PDE1, PDE2].

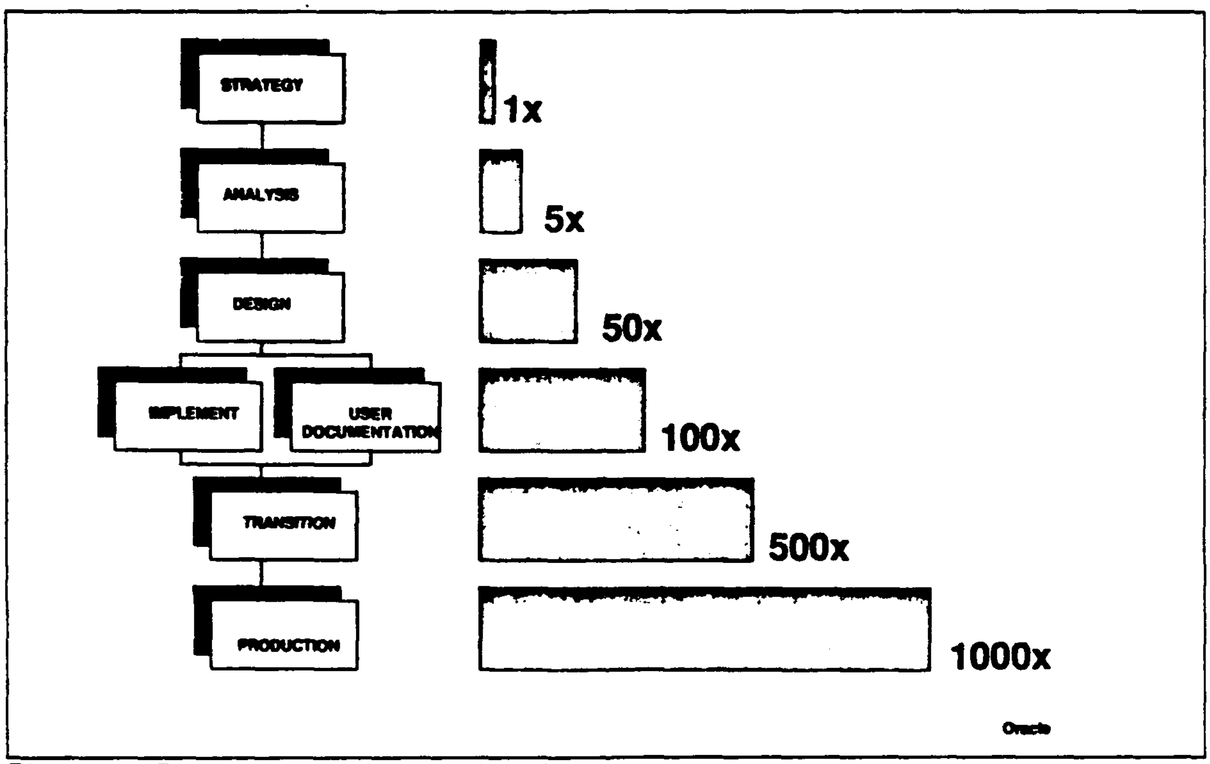


Figure 1 Relative cost of correcting defects

Research in software engineering has shown that the most costly errors to correct can be traced to the specification and design stages of development [Het88, Mye76]. The cost to correct such problems can be spectacularly high (see Figure 1). Reports on methods to improve software quality state that 25% of the effort on a project should be allocated to verification and testing [Het88]. Furthermore, these activities should be started as early in the development process as possible. The definition of what to test should begin when the requirements are specified.

---

[3] For a discussion of accepted software testing practices refer to The Complete Guide to Software Testing, by B. Hetzel [Het88].

[4] This term and others used in this paper are defined in Section VI.

Verification (through inspection) should occur at every stage and validation should be used at the end of every prototype phase [Bah91].

The software quality and knowledge engineering methods related to verification and validation are useful for STEP, up to a point. The principles found in them are valid but the existing techniques are not generally transferable to STEP. Validation techniques that are appropriate for the methods used by STEP need to be defined, accepted and implemented.

Testing early in the STEP development process benefits both STEP developers and STEP implementors by providing for:

1. earlier discovery of defects (permitting corrections with less effort),
2. fewer defects in completed products,
3. increased developer productivity,
4. increased design efficiency,
5. reduced development time, and
6. improved testability.

Status within the STEP Community

Validation is currently a recommended practice in the AP Guidelines [WG4-N15] but is not required. Before STEP is standardized, the development process should ensure that STEP is relatively free of errors. Because of the coordination required to reach international consensus, there are inherent delays in correcting defects in standards. STEP AP projects should be required to provide the evidence that a workable solution has been put forward. This evidence should be in the form of a validation report that is sufficient to convince experts of the correctness of the specification. However, a validation report is presently an optional part of an AP specification. The validation report should be included when the AP is submitted to ISO for elevation to committee draft status. AP development projects will need adequate resources to support a requirement for a validation report.

Quality reviews and inspections have been incorporated into the STEP development process [IPO1, WG4-15]. The inspection process performed by the qualification project in the Qualification and Integration Working Group (ISO TC184/SC4/WG4) has improved the STEP draft specifications. Indeed, research in software testing has determined that inspection requires a fraction of the effort of full implementation testing to locate and correct defects over the normal life cycle of a software product [Fag86, Fre85]. However, inspection techniques alone are not sufficient. The techniques used by WG4 inspect the STEP specifications for specific qualities but cannot evaluate the correctness and useability of the technical content of models. The STEP information models can be validated only by testing how well they support the functional requirements described in Application Protocols. The ability to trace validation tests to the requirements provided in the AP is critical to successful validation.

When to Test STEP

In the report "Considerations for the development and implementation of PDES within a Government Environment" [Hen89], three goals of pre-standardization testing are stated:

- demonstrate completeness and correctness of the descriptions,
- demonstrate that the STEP specification can support a working solution,
- demonstrate an implementation technology.

The validation methodology which is being proposed in this paper will satisfy the first two of these three goals. The third will require an actual implementation.

There are at least two points in a system development process at which the system can be evaluated to assure that it achieves all functional requirements: 1) evaluate if the design meets the system requirements or 2) evaluate if an implementation of the design meets the system requirements. In Figure 2 these two points are shown at stages 2 and 3. The methodology described in the next section provides for the evaluation of the design of the AP. Evaluation of prototype implementations of APs (choice 2 above) will be expensive in both time and money and should be postponed until after the Application Protocol has been validated.

The proposed approach is to validate APs by simulating the behavior of the relevant application. Simulation techniques have promise for validating proposed standards. Other standards efforts using formal descriptive techniques have found it essential to build a simulation environment [Sij89]. There are a number of specific techniques that could be used to perform this simulation. The choice of a specific set of techniques should depend upon the software tools that exist to reduce the manpower requirements of the testing and the skill levels of those performing the tests.

The validation tests are identified by examining the functions of the application. The data required to perform each activity in the process is specified in detail. Realistic data from the application domain is associated with each of these tests. The data needed to perform a specific activity is then associated with the structures provided by the application model. Then the application model with its associated data is examined to determine if it can support the generation of the required outputs for the process. The method is essentially simulating the behavior of an application system interacting with a user of the system.

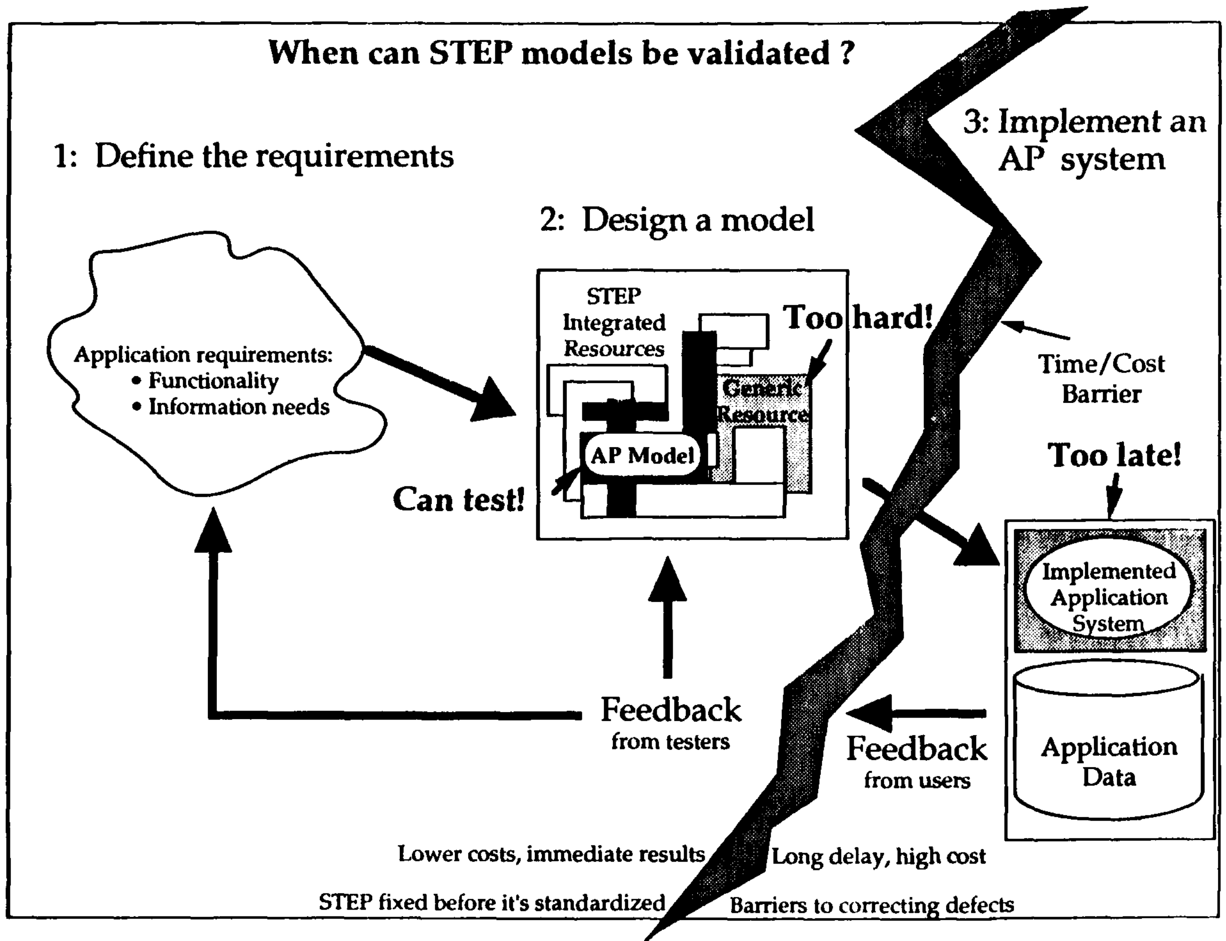


Figure 2 Opportunities for testing

Verification and validation of an AP requires experts in the domain of the application. Obtaining application expert commitment and participation is critical. Independent review is necessary for verification and validation to provide the best possible results. The experts which verify the knowledge representation and accept the testing results cannot be the same group that is responsible for producing the AP or defining and executing the validation tests. Using experts from a number of representative industries will improve the quality and workability of the AP.

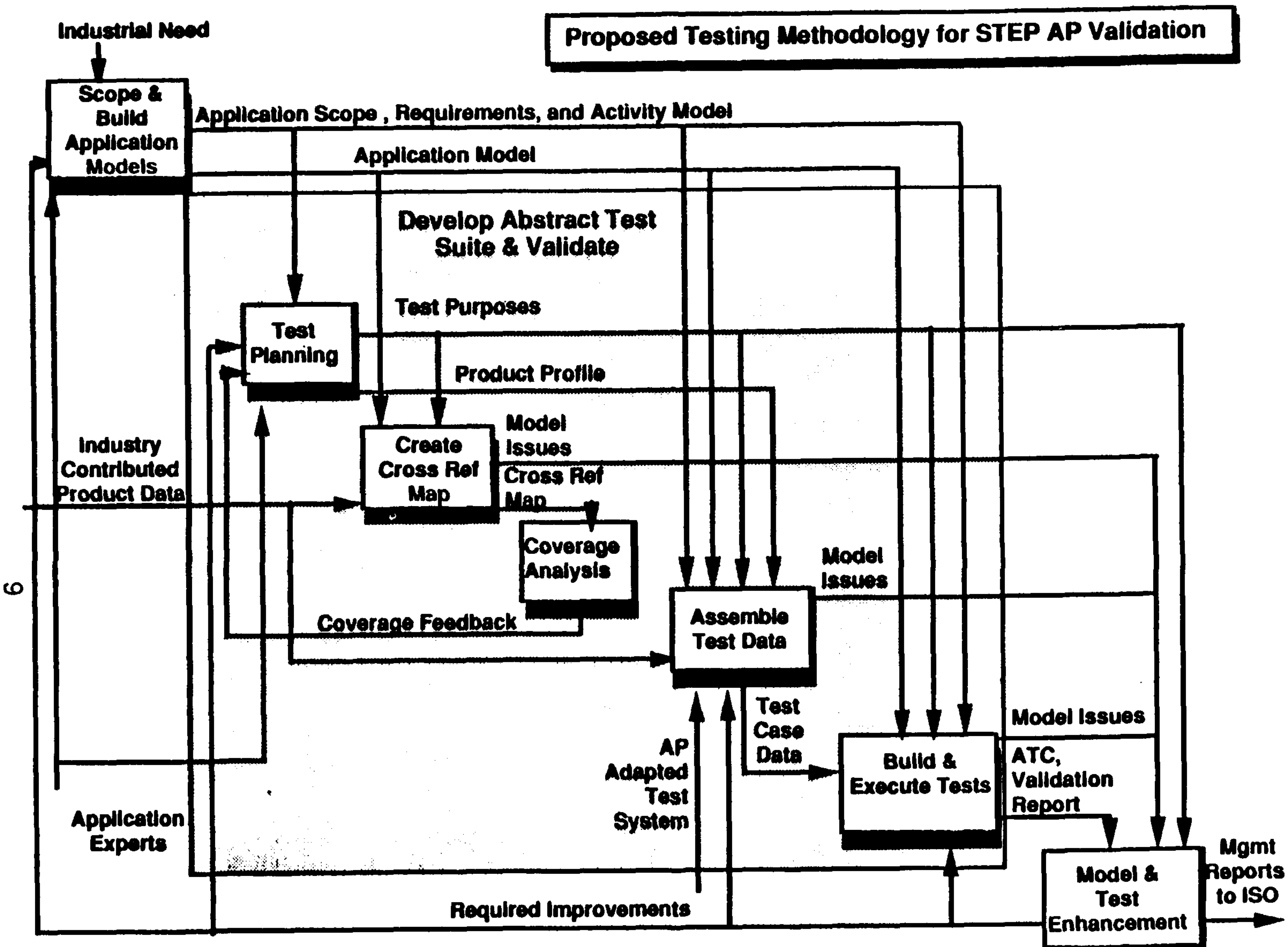
III. Validation Testing Methodology

This section presents the major aspects of the proposed testing methodology for STEP application protocol validation. These aspects can be broken into two major sets of functional activities. These are: 1) planning activities and 2) testing activities. An overview of the validation testing process and the most significant inputs and outputs from it is illustrated in Figure 3.

Test planning is constrained by the scope and requirements of the particular application. In addition, this activity results in the creation of test purposes and the identification of test data. A test purpose is a general statement of the intent of each test. Tests become progressively more detailed until completely specified tests are developed. Each test is exercised against the application model of the AP. The complete set of test specifications for one AP is called an abstract test suite. The abstract test suite for an AP will be a companion standard to the AP and is required for the conformance testing of that AP. Therefore, it is recommended that the developers on an AP also develop the *abstract test suite*.

Validation testing, the second major set of activities in this testing methodology, is when the actual validation of the AP is performed. Test feedback, which ensures that any defects in the draft specification are corrected, is provided during these activities. During the validation testing activities a cross reference map is created, test data is assembled into test cases, and the tests are executed and analyzed.

For each activity, the following format has been used to present the methodology: 1) the activity is identified, 2) the purpose or objective is stated, and 3) a detailed description of the actions to be performed is provided. Relevant examples from one STEP application protocol Configuration Controlled Design [WG4-N14], are used throughout. Readers who are interested in understanding a fully elaborated AP should read the Configuration Controlled Design AP document [WG4-N14]. At the end of this section, the major deliverables from each activity and the opportunities for automating portions of the activities are presented in the format of Tables.



*ATC = Abstract Test Cases

Figure 3 Proposed methodology for STEP AP validation

3.1 Planning Activities

In this sub-section, the two activities associated with planning the tests are discussed. These activities are: 1) develop a test plan with test purposes and 2) gather test data. The planning activities require a precise definition of the intended scope of the application protocol (see Appendix A for a complete discussion). However, the test planning can proceed without having an application model (as shown in Figure 3). The application models in an AP specify the detailed information requirements of the application (sub-section 3.2 provides a detailed discussion). Once these two activities are completed, the activities which validate the application models can commence.

1) Develop a test plan with test purposes - identify and organize the collection of tests that are needed to validate an application model.

The test plan provides a high-level description of what the testing will cover. It is used to determine if sufficient information is contained within the application model to support the application requirements. A test plan is developed to identify: a) the testing strategy, b) the aspects of the application model that are to be validated, and c) the sequence of tests to be performed.

In the test plan, the testing requirements are specified in significant combinations called test groups. Each test group is associated with a unit-of-

Units-of-functionality (UOF) for AP 203, Configuration Controlled Design:

- **Assembly Component Structure:** Identifies the relationship of components in a conventional engineering assembly.
- **Design Change:** Provides the ability to manage the proposal of, the approval of, and implementation of a change to a product design.
- **Effectivity:** Identifies an intended physical manifestation of a product for the purpose of configuration management of one or more physical units.
- **Shape:** A definition of the size, spatial configuration and proportions of a real or conceived product or part which associates a particular type of geometric representation to a product design.
- **Wireframe Geometry:** A type of geometric representation for defining shape which contains only geometric curves, often referred to as unstructured geometry, and contains no topological information.
- **... other UOFs**

Figure 4 Test group organization by UOF functionality (UOF). A unit-of-functionality describes one specific category of behavior that the application must support to satisfy the accepted practices of the application domain. A test group is organized by unit-of-functionality to improve its readability to aid the experts from the application domain. These experts then verify the contents of the test group. A partial list of the units-of-functionality from our example application protocol can be found in Figure 4. The tests within a test group may need to be performed in a particular order.

Each test group has a set of test purposes associated with it. A test purpose is used to specify some characteristic which must be present in the application model of the application protocol. Normally, an application expert would participate in defining test purposes. An example of a basic test purpose might be: "Does the product definition include a product version identifier?"

Test purposes are divided into two categories, basic and complex (see Figure 5). Basic test purposes are used to either: a) evaluate a single characteristic of one

Test Group: Assembly Component Structure

1. Basic Test Purposes:
    a. Product must be a part. No tests are needed for product, just for part:
        1 product as part with description present
        2 product as part with class as assembly
    b. Product_version is a revision of exactly one product (part). Product_version tests:
        1 product_version with description present

...

2. Complex Test Purposes:
    a. Given an assembly product_definition, determine which component parts are standard parts.
    b. Given a project identifier, determine the highest assembly level parts list.
    c. Given a product_definition, determine the product_version of each (approved) substitute.

Figure 5 Examples of test purposes

construct, e.g., an attribute of an entity defined in Express, or b) evaluate the relationship between two or more constructs. A complex test purpose is used to evaluate whether or not the application model can support the necessary usage defined by the application experts. The test purposes provide the basis for defining abstract test cases, described in Section 3.2 below, the basis for conformance testing requirements for the AP.

Each application protocol may target a limited set of product families. Our example application protocol is intended to support only mechanical parts or rigid assemblies. Characteristics of the product families supported by the AP must be specified. This information is used to determine what test data will be required to adequately test the AP. For this reason, the characteristics of the product families are documented in a "product profile". Figure 6 provides a simplified example of a product profile. It is used to guide the selection of example test data. Multiple sets of test product data are often required to provide a representative data sample for the stated application requirements. As there may be many ways to accomplish the same function within an application, a single set of test data may not reveal the most relevant uses of the data.

Product Data Characteristics:

1. mechanical part with
   a. shape defined by wireframe geometry
   b. wireframe geometry must include simple and complex curves and simple conics
2. rigid assembly with
   a. assembly design defined with surface geometry and bound with topology
   b. surface geometry must include surfaces of revolution and complex curve surfaces
   c. topology must include edges, edge loops and faces

Figure 6 Product profile for AP 203

When the planning activity is completed, the following items will exist:

- a test plan document with itemized test purposes, and
- a product profile which identifies relevant test product characteristics.

Experts from the application area should review and approve each of these items before the next activity. These experts are also invaluable in the next activity to identify existing industry data repositories.

2) Gather the test data - locate, organize, and record the test product data needed to meet the testing requirements.

This activity uses the product profile and test purposes from the last activity to locate the product data that will be used in validating the application model. Actual product data resides in industry in a number of both manual and automated representations. The data could be represented by engineering design drawings and documentation, archival files from IGES representations, or files extracted from computer-aided engineering systems.

There are two reasons for acquiring actual product data from industry. First, this reduces the need to check the validity of the data prior to validating the application model. Second, test data is more likely to have the set of conditions required to ensure that the semantics in the application model meet the application requriements.

The documents and files are given an identifier. The person responsible for this task then goes through a process of selection and reduction. Specific pieces of information that satisfy a specific test purpose are identified. The same item of test data may be used to satisfy one or more test purposes. Acceptable ranges of data values should also be identified. Each piece of test data that is identified is labeled and verified against the application requirements.

This is an internal activity to organize and record the test data that has been identified. Table 1 is provided to illustrate a report that might be provided for use in the formal testing activities described in the next section. As an example, the Test Purpose 1a.2 has an ARM ENTITY PART with Attribute part_class. Data for this test purpose is stored in the files named LOCK1.DOC/3 and CHAS1.FIL/15.

Test Purpose	**ARM ENTITY/ Attribute**	**SOURCE/Test Data Identifier**
ASSEMBLY COMPONENT STRUCTURE		
1a.1 description present	PART->DESIGN_ PRODUCT_DEFINITION /description	LOCK1.DOC/2 CHAS1.FIL/8
1a.2 class is assembly	PART/part_class	LOCK1.DOC/3 CHAS1.FIL/15
2a.1 product is assembly	PART/part_class-> ENGINEERING_ ASSEMBLY	LOCK1.DOC/3 CHAS1.FIL/15
2a.2 assembly has components of quantity	NEXT_HIGHER_ ASSEMBLY/ component_quantity	LOCK1.MRP/5,6,7 CHAS1.FIL/20,24,26
2a.3 components are standard parts	PART/standard_part _indicator	LOCK1.MRP/6

Table 1 Data sources for the test purposes in Figure 5.

3.2 Testing Activities

In this section, the major activities of the validation testing process are discussed. These include: 1) create cross reference map, 2) perform coverage analysis, 3) assemble test cases, 4) develop test cases and execute, and 5) manage feedback and refinements. The first four are actual testing activities and the last is an iterative approach for resolving the issues uncovered by testing.

Validation testing activities require additional portions of the application protocol than were needed for the planning activities. These are: a) application reference model (ARM) which specifies the information requirements of the application, b) the application interpreted model (AIM) which applies STEP standardized concepts to satisfy the requirements of the application, and c) the ARM to AIM mapping report which relates the ARM to the AIM (see Appendix A for a complete discussion). In the following sections the term application model is used to refer to either the ARM or the AIM.

This validation testing process could be used to validate either of these components of the application protocol. However, the AIM is most critical as it is what will be implemented. The effort required to apply the full set of validation testing activities to the ARM is better spent on the validation of the AIM.Therefore, the full process will be applied to the AIM. As the ARM is less critical, only the cross reference and coverage analysis activities will be applied to the ARM. These activities are likely to uncover any major flaws in the ARM or any deficiencies in the coverage of the test data.

Validation testing of an AP can be accomplished without taking the extra step of preparing formal test specifications. These specifications, or abstract test cases, are the basis for the abstract test suite that will be required for conformance testing of the AP [ISO31]. Even though abstract test case development is not currently required by the AP guidelines [WG4-N15], providing them as a by-product of the AP testing has many advantages. As the abstract test cases are also constructed against the test purposes developed during test planning, validation test personnel will develop the required knowledge of the AP during the first activity discussed below. In addition, validation personnel can use these formal specifications: a) to evaluate the impact of changes to the application model that is under test, and b) as a control on alternative methods of building executable tests. Activities 3 and 4 are used to produce abstract test cases for conformance testing. The development of formal test specifications requires a controlled environment so that each abstract test case will be complete, consistent with the application model, and traceable to requirements. The draft document that will specify the format and content of abstract test cases is not sufficiently complete to include specifics in this proposal [WG6-N15].

1) Create cross reference map - determine how well the application model and the test data match each other.

This activity can begin when the application model is released for validation by its developers. The personnel performing this activity must first gain a thorough understanding of the entire application model. The initial objective is to locate where the gathered test data should reside in the model. This is done by examining the Express constructs in the AIM or the information model in the ARM (see Appendix A for examples of each). Included are the associated definitions and the test data gathered for each test purpose described in the test plan. This analysis requires two steps. First, the relationships between the test data and the application entities are examined. Second, each specific piece of data is associated with an attribute within the entity. All of the identified data should have a logical home within the model. Data which cannot be tied to a specific construct may mean that the model is incomplete.

It is not necessary to use every piece of the gathered product data to validate the application model. Data is selected that meet the unique combination of conditions that are specified in the test purpose. For example, if the shape definition for a product contained eight curves with very similar characteristics, only one would be needed by the cross reference activity.

There should be only one way to match each piece of test data with an attribute of the model for a specific test purpose. This is true if the test purposes are directly traceable to the data structures in the AIM or ARM. This activity is a major portion of the validation testing.

Past experience with testing [PDE2] has shown that at least half of the deficiencies uncovered in validation testing are found by this activity. Any discrepancies uncovered as a cross reference mapping is developed are documented as issues against the application protocol.

2) Perform coverage analysis - identify unused parts of the application model.

After the cross reference mapping activity is completed, a systematic review is performed to determine how much of the application model can be examined with the actual test data. In the first pass, entities are identified that have no test case data at all. In Figure 7, a subset from our example application model (AP 203), is represented in Express-G [ISO11], a graphical representation of Express, and the entity with no test data is shaded.

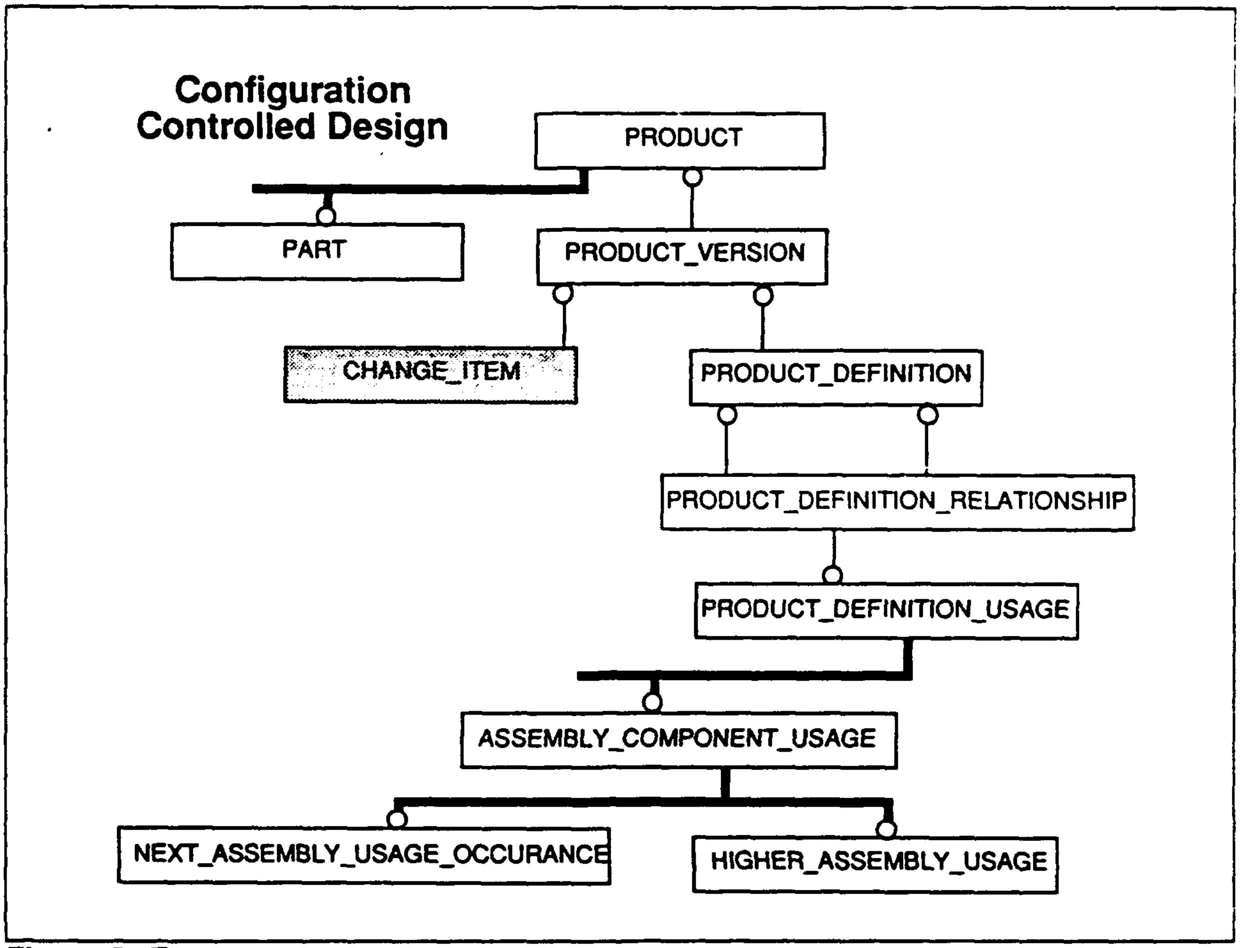


Figure 7 Test case coverage analysis

Once these voids have been identified, another attempt to gather test data for these test purposes is made. Any requirements which can not be verified are removed from the application model. In a second pass through the model, structural coverage is identified. This checks that all paths between entities are utilized and that there are not multiple paths that satisfy the same requirement. Multiple paths mean that there is either redundancy that should be eliminated from the application model or ambiguity that may cause the requirements to be refined. The analysis may also determine if additional test purposes are needed to cover the application model.

NOTE: Establishing a test system is a precursor step to the remaining testing activities. Therefore, the use of an automated system configured to the application model under test is assumed. This requires that the application model be checked for syntactic correctness with Express and that version control be established. The primary output is the test system that has been configured for the specific AP which is under test. The software which configures the test system should be evaluated to ensure that the Express constructs are adequately represented. In addition, a system log records any messages generated by the Express-based software tools. This step is summarized in the following table:

Administrative step	Required input	Generated outputs
Establish test system	Application model in Express	1) Test system configured for application under test 2) System log

Table 2 Establish test system

3) Assemble test cases - evaluate if certain test purposes are met by the application model and if the application model data structures can support realistic data.

During this activity, the application model is populated with the actual data contributed from industry. The data is stored in a test system that supports the constructs in the application model. There should be only one way to populate an attribute of the model withe each piece of test data for a specific test purpose. The Express construct is appropriate if: a) its definition is consistent with the usage from the test purpose and b) the structure fits the need, e. g. if three coordinates are needed to locate a point, the Express construct specifies exactly three coordinates.

There are a number of methods and some tools available that could be used to complete this task. A relational database could be populated by using SQL[5] commands. Another method is to create STEP exchange files and build a test system that can import or export these files. Translation programs can be used to extract data from existing automated systems and format it into STEP exchange files. Finally, a STEP-based editor could be built which would lead a tester through the task during an interactive session.

The PDES, Inc. testing activities (see Appendix B for more information) use a combination of all of these methods, based on the availability of the data in a computerized form. Once populated, having the data in a persistent database has definite efficiency advantages for this and following testing tasks.

Once the application model has been populated, the test case data should be organized by test group to facilitate the management of the test data. For example, if

[5] American National Standard for Information Systems - Database Language - SQL is American National Standards Institute, Inc. X.3.125-1986. SQL is a database language for the schema definition and data manipulation of relational databases.

a database is used, each test case would be identified with its test group by an identifier file within the database.

These computerized forms of the test data are the executable part of an executable test case. The abstract test case necessary for conformance testing requires that the same logic and test data to be specified, but it will not contain the details of a particular test system.

Careful analysis of the difficulties uncovered during data population is necessary to ensure that the problem is caused by the application model under test. Some problems may result from the interpretation of the test purposes, or from errors in the test data or the test system.

4) Develop test cases and execute - evaluate if the more complex, usage specific test purposes are met by the application model.

Once a test system has been populated, any remaining test purposes can be fully specified. This testing activity is intended to address the following concerns. Can the outputs of an application process be generated with the data in the test system? Does the meaning remain intact and do data structures from the application model under test support a single way to access the data?

Queries are written and executed to cover these usage specific test purposes. The queries are written to reflect real world questions which an enterprise would need to answer that are within the scope of the AP. The results of these queries can thus be readily verified by application experts.

The queries will be defined against a specific implementation of the test system. If a relational database was used they would be written in standard SQL. Other test systems might require them to be written in application code.

Successful execution of the queries proves that there is a correct access path to the data and that correct data relationships exist in the application model under test. There should only be one path in the application model for a specific test purpose. Queries that do not produce the expected result will cause issues to be generated against the AP.

The logic and the resulting test case data from the executable query are used to generate the abstract test cases for the remaining test purposes.

5) Manage feedback and refinements - ensure that the validation of the application model is completed by a) tracking issues, b) determining requirements for re-testing or other changes, and c) re-executing the required test cases.

During this activity the results of the executed tests are analyzed and documented. The four previous activities are likely to produce issues against the AP but they may also identify deficiencies in the test purposes, test suite, or test system. Some portion of the issues are likely to impact other STEP models (see "Integrated Resources" in Appendix A). A five step process for proposing improvements to the models and for managing the testing process evaluation of these improvements follows:

a) The tests should be organized into groups of related tests, called test groups. Each test should establish clear criteria so that results can be judged either pass or fail.

b) At regular intervals the issues should be collected from the four previous testing activities and consolidated into a single document. The most efficient method would be to execute, analyze and report on all tests in one testing cycle. But serious defects are likely to impact other test cases and require extensive effort by the model developers to correct. PDES, Inc. found that a practical interval was four to six weeks. The individuals who executed and analyzed the tests will need to review the issues with the developers of the application model. The disposition of each issue should be assigned during this review. Issues that impact STEP Integrated Resources will need to be forwarded to ISO and reviewed there as well. The ISO projects may formulate an alternative solution to an issue so these must be tracked and tested. Additional effort by the model testers is now required to understand a revised application model and to modify the test cases to conform to new model. The experience from the PDES, Inc. testing was that each test cycle found about half the number of defects found in the previous testing cycle.

c) As issues are resolved and incorporated into the application model, the resolutions should be documented as a supplement to the application model to provide visibility to the changes. Validation testing can then be limited to these updated areas of the AP model. The revised application model is released at some fixed interval of time.

d) A new test system is generated that conforms to the revised application model. Additional test purposes may need to be specified and data gathered to support them. Test case data may need conversion to be compatible with the revised model [Koh90]. The abstract test case specifications may also need to be modified to reflect these changes.

e) The new test system is loaded with data. Re-testing of previously executed test cases determines if they are impacted by changes in the application model.

The entire process is repeated until: a) there are no remaining problems, e.g., all issues are resolved and have been successfully tested, or b) the modelers feel that they cannot further improve on the model. It is expected that three testing-cycles will be needed to have confidence that the most serious defects have been located. The AP validation report can then be completed. A workshop should be conducted to review and accept the validation. The participants in this workshop should include application experts who were not responsible for producing the validation report.

In Table 3, the results of the validation testing activities are identified along with a reference to the accepted representation format.

Activity	Generated outputs	Representation
Develop test plan	1) Test plan 2) Test purposes 3) Product profile	Text document Per AP Guidelines [WG4-N15] Text document
Gather test data	1) Table of test purpose and data sources	Text matrix or table
Create cross reference map	1) Cross reference map 2) Model issues	Text table, Express-I [WG5-N19], or Express-G Issue document per SC4 directives
Perform coverage analysis	1) Test coverage report 2) Test matrix	Text document Text matrix or Express-G diagram
Assemble test cases	1) Instance transactions 2) STEP file if file exchange AP 3) Model issues	Compatible with test system ISO 10303 Part 21, STEP exchange file Per SC4 directives
Develop test cases and execute	1) Abstract test cases 2) Executable test cases 3) Test logs 4) Model issues	Per ISO 10303 Part 33 Compatible with test system System log Per SC4 directives
Manage feedback and refinements	1) Validation test report 2) Model issue resolutions 3) Revised application model 4) Revised test plan and test purposes 5) Revised abstract and executable test cases 6) Deficiency reports against testing software	Per AP guidelines Per SC4 directives Express model Text document and per AP guidelines Per ISO 10303 Part 33 and compatible with test system Text document

Table 3 Outputs from the validation process

Table 4 provides a discussion of the software functionality that would aid in performing each activity within the proposed validation testing methodology. The requirements for the test system discussed in Table 2 have been included at the appropriate point in the sequence of activities.

Activity	Automation requirements
Develop test plan	A document processing system with a cross referencing capability is needed to generate the test plans and test purposes. The ability to browse the graphic representations of the AAM and supporting documentation is useful for deciding what needs to be tested.
Gather test data	A simple table management capability with a basic sorting and string search ability is sufficient.
Create cross reference map	The ability to browse both the AP and STEP Integrated Resources documentation including the Express models is needed. Formal SGML[6] tagging of these documents could be useful for browsing and to improve traceability. Software that could generate Express-G and allow labels to be attached would also be useful.
Perform coverage analysis	None. Once the tests are fully and formally specified, automation would be possible.
Establish test system	An Express parser is needed that can process the application model and then generate or configure a test system for the application model under test.
Assemble test cases	Translators for IGES, CADx systems, and other automated systems that convert product data into a STEP exchange files are needed. STEP exchange file import and export software is needed for the test system. The test system must support the merging of multiple STEP exchange files. A full range of editing features must be supported. During editing sessions, consistency and completeness checking need to be under operator control. Configuration management and access control are also needed.
Develop test cases and execute	A test session manager that would log the configuration of the test system, the specific executable tests that were run, and capture and store the test output would aid the analysis of the test results.
Manage feedback and refinements	A data conversion capability is needed to reformat test data to match any changes to the application model's Express specification. Configuration management is needed for the application models, the STEP Integrated Resource models, test purposes, STEP exchange files, and test cases.

Table 4 Description of automation requirements for test activities

[6] Standard Generalized Markup Language (SGML) is ISO standard 8879 for document markup. ISO is in the process of placing all of its standards in this format.

IV. Relationship of Validation Testing to STEP Conformance Testing

Conformance testing evaluates the extent to which an implemented application system complies and conforms to the requirements specified by a standard. A standard is only as good as the conformance tests and the independent testing program which ensure that vendor products actually implement the standard.

Validation of an AP can and should provide a foundation for the conformance testing of implementations that claim to implement the AP. The objectives of each type of testing are different, but the end products of validation testing can be used to provide necessary inputs for conformance testing. Two Parts of the proposed STEP standard that relate to AP validation and conformance testing are: Part 31, "Conformance Testing Methodology and Framework: General Concepts" [ISO31], and Part 33, "Structure and Development of Abstract Test Suites" [WG6-N15]. These Parts state the requirements for conformance testing. The approach to testing and what is tested does not need to be different because the requirements for both validation testing and conformance testing are very similar.

The process of validation identifies the tests which are needed to satisfy the functional requirements. These same tests are required for conformance testing. The other major aspect of conformance testing is to ensure that the implementation under test will correctly deal with unacceptable conditions, called falsification testing. The existence of tests which evaluate functionality make falsification tests much easier to specify. However, falsification tests are not produced by the validation testing activities.

The test data developed for AP validation can be used in the abstract test cases required for conformance testing. The team of experts that has validated an AP will thoroughly understand the AP and the requirements that drive it. Thus this team is best equipped to design the abstract testing suite that will be used in validation and conformance testing. The organization that provides the team to validate an AP could potentially recoup some small portion of the costs by licensing any executable test cases that were produced during the validation testing activities. These are just some of the efficiency and quality reasons for defining detailed conformance testing requirements as part of an AP validation process.

Within the STEP community, it has not yet been determined which projects will be responsible for validating application protocols and which development projects will fully specify the abstract test suite. The industry that needs the AP has the most motivation for accepting both of these responsibilities [ITI91]. Proof that an industry need exists is critical to getting an AP development project recognized within ISO.

Currently, an AP development project is encouraged to validate its application model but the project is not responsible for specifying the abstract test suite. But the

AP project must specify the requirements for validation in the test purposes which are needed for both validation and conformance testing. The AP project has both the necessary technical skills and the most motivation for performing both of these activities.

V. Conclusion

Verifying and validating STEP specifications prior to standardization are cost effective methods of ensuring that STEP is free of technical flaws. Validation of application protocols is critical to ensure that the standards will provide practical and useful specifications for STEP implementations. Incorporating this strategy into the STEP development process benefits both STEP developers and STEP implementors by providing for:

1. earlier discovery of defects (permitting corrections with less effort),
2. fewer defects in completed products,
3. increased developer productivity,
4. increased design efficiency,
5. reduced development time, and
6. improved testability.

The method and techniques proposed have been evaluated in previous testing of STEP with good results. In recent years, scores of model issues were identified against STEP resource models [PDE2] that were thought to be technically complete in 1988. Some of the methods required to develop application protocols are so new that they are still undocumented. As the first STEP APs emerge, these techniques will be tried. The proposed validation method has already been informally accepted within the STEP Community. In the long run, application protocol validation will save time and effort. The improvements that validation testing can make on STEP are measurable, as demonstrated by early testing activities.

The AP projects are best equipped to carry out the validation. These projects are more likely than other STEP projects to have direct funding for a fixed duration. Therefore, AP projects are more receptive to ideas concerning how to produce a quality AP the first time. The most appropriate forum for solidifying these techniques is probably the Integration and Qualification Working Group (WG4) in TC184/SC4. Validation is a logical extension to their efforts to require that all STEP models be qualified. These efforts are currently applied by using a manual process of document verification. A project within WG4 should be established to pursue the adoption of AP validation in STEP development. Liaison activities will be needed with the Conformance Testing Working Group (WG6). This will ensure that the results of validation testing can provide the basis for the abstract testing suites needed for STEP conformance testing.

The foundation for pre-standardization testing is derived from software implementation testing. The proven verification and validation principles for software testing are valid for testing STEP application protocols. This report proposes a methodology and supporting techniques that are appropriate for the methods used by STEP. Some verification techniques are used by WG4 and are required in the development of STEP. These techniques require some further refinement for use with application protocols. A specific set of validation techniques needs to be defined, accepted and implemented by STEP.

Validation of APs can and should provide a foundation for the conformance testing of STEP implementations. The approach to testing and what is tested need not be different as the requirements for both validation testing and conformance testing are very similar. The validation process identifies the tests which are needed to satisfy the functional requirements. The existence of documented tests for what is acceptable functionality will make specifying tests for unacceptable behavior, or falsification testing, much easier. Furthermore, the test data and test specifications developed for AP validation can be re-used in the abstract test cases for conformance testing. A standard is only as good as the conformance tests and the independent testing program which ensures that vendor products actually implement the standard.

VI. Terminology

Abstract Test Case: A complete and implementation independent specification of the actions required to achieve a specific test purpose.

Abstract Test Suite: A complete set of abstract test cases, organized into test groups, that is necessary to perform conformance testing for a standard Application Protocol.

Application: An enterprise process that puts product data to use. The scope of an application is defined by the class of product, the supported stages in the life cycle of the product, the uses of the product data, and the disciplines that use the product data.

Application Activity Model (AAM): A representation of one or more activities which use product data in a specific application context. An AAM is used to establish understanding and agreement of the application activities and processes.

Application Interpreted Model (AIM): A model that describes the interpretation, through the selection and addition of constraints, of the STEP Integrated Resource constructs. The interpreted constructs that result provide functional equivalence to an AP's information requirements.

Application Protocol (AP): A standard that specifies implementable STEP constructs. It defines the context for the use of product data and specifies the use of STEP to satisfy an industrial need.

Application Protocol Validation: The process of evaluating a candidate AP and its components to determine whether these satisfy the specified requirements.

Application Reference Model (ARM): A model that formally describes the information requirements and constraints for an application domain. The model uses application-specific terminology and rules familiar to an expert from the application domain. The model is independent of any physical implementation.

Conformance Testing: Testing the extent to which an implementation under test is a conforming implementation.

Construct: A logical grouping of concepts based on meaning. A construct is conceptual.

Executable Test Case: An executable test case is derived from an abstract test case (ATC) by assigning values to the parameters of the ATC and then building a test program per other instructions in the ATC. In conformance testing, this test program is then executed against the implementation under test.

Executable Test Suite: A complete set of executable test cases that has been built according to the specifications in an abstract test suite specification. The test suite is organized into test groups and it provides instructions that guide the execution of the test suite.

Falsification Testing: A test method developed to find errors in an implementation. Test cases are developed which intentionally do not meet the specifications for conformance with a standard. If the implementation does not detect these errors, one can deduce that the implementation does not conform to the standard. However, the absence of detected errors does not imply conformance. Test cases which examine conformance are also required.

Fitness Testing: The peer review and walk-through of a model which demonstrates that the model is useful in a particular application domain.

Functionality: The specified capability that must be provided to meet the requirements of a user(s) of a system(s).

Integrated Resources: Those parts of STEP which provide the structures that carry the meaning of product data in its most broad context, e.g. across the product life cycle and across manufacturing disciplines. Within this set of parts, there is no redundancy and they are consistent throughout. For STEP Version 1.0, these include: Integrated Resource Fundamental Concepts, General Shape Representation Concepts (which includes geometry and topology), Representation Structures, Product Structure Configuration Management, Presentation, and General Draughting Concepts.

Integrity Testing: Those tests which demonstrate that a model is syntactically correct and self-consistent.

Model Issues: Documents a problem or concern about an aspect of an information model, includes a textual description and proposes alternative resolutions.

Product Profile: The characteristics that describe a specific type or category of products or parts.

Product Data: The set of data elements that is necessary to fully support a product and all its in-service needs over its expected life cycle. The set of data elements includes all the data needed to completely define the product, plus other data pertaining to the operation and maintenance of the product until it is removed from service.

STEP Integrated Resources: See Integrated Resources.

Test Case Specification: A document specifying inputs, predicted results, and a set of execution conditions for a test item [ISO1].

Test Group: A named set of related tests that have a common objective which the test purposes within a specific test group are designed to achieve [ISO1].

Test Log: A document that records the initial state of the test system, the sequence of all tests executed in a test session, and the results of each test.

Test Plan: A document that describes the overall strategy and sequence of tests that are required to evaluate the functionality of an application model.

Test Purpose: A description of a narrowly defined objective of testing, focusing on a single conformance requirement as specified in the AP that is being tested [ISO1].

Test System: An automated system which has been built to embody the data structures and properties of the information model under test.

Verification: An inspection process which ensures that a component, such as an ARM or an AIM, is technically correct [Bah91].

Validation: The process of evaluating a system or component to ensure that the "right" system was built. Validation determines whether the component or system does what it was intended to do and whether it satisfies the specified requirements. It determines the correctness of the system or component [Bah91].

Validation Report: A document which summarizes and records the results of the validation process.

Validation Testing: The process of developing, executing and evaluating test cases to explore the system or component and expose errors [Bah91].

Unit of Functionality (UOF): UOFs are documented as textual descriptions in the ARM of an Application Protocol. A UOF is a construct that performs one specific function according to the accepted practices of the application. The ARM is organized by UOFs to improve its readability to aid the experts in the application domain.

VII. References

[Bah91] Bahill, A.T., Verifying and Validating Personal Computer-based Expert Systems, Prentice Hall, Englewood Cliffs, NJ, 1991.

[Bre91] Breese, N., McLay, M. and Silvernale, G., "Validation Testing Laboratory User's Guide," NISTIR 4683, National Institute of Standards and Technology, August 1991.

[CAL90] "Graphics Communications between Different Computer Systems: A Review of IGES Translators, their Implementation, and Recommendations," Computer-aided Acquisition and Logistic Support (CALS) Army Program Office, Fort Mommoth, NJ, June 1990.

[Dan91] Danner, W.F., Sanford, D.T., and Yang, Y., "STEP Resource Integration: Semantic and Syntactic Rules," NISTIR 4528, National Institute of Standards and Technology, March 1991.

[Fag86] Fagan, M., "Advances in Software Inspections," IEEE Transactions on Software Engineering, July 1986, Vol SE-12 Nbr 7 pg 744-751.

[Fre85] Frewin, E., Hamer, P., Ketchenham, B., Ross, N., and Wood, L., "Quality Measurement and Modeling-State of the Art Report," ESPRIT Report, Request STC-gdf/001/51/QL-RP/00.7, July 1985.

[Hen89] Henghold, W., Shumaker, G., and Baker, L., "Considerations for the Development and Implementation of PDES within a Government Environment," Interim Report, AFWAL-TR-89-8009, Air Force Systems Command, Wright-Patterson Air Force Base, OH, February 1989.

[Het88] Hetzel, B., The Complete Guide to Software Testing, Second Edition, QED Information Sciences, Inc., Wellesley, MA, 1988.

[ICA82] ICAM Architecture Part II, Volume IV, Function Modeling Manual (IDEF), Air Force Systems Command, Wright-Patterson Air Force Base, OH, February 1982.

[ICA85] Information Modeling Manual IDEF Extended (IDEF1X), ICAM Project Report (Priority 6201), Air Force Systems Command, Wright-Patterson Air Force Base, OH, December 1985.

[IPO1] Jorgensen, B., Tyler, J., Williams, A. and Zimmerman, J., "Model Qualification Guidelines for PDES/STEP," IGES/PDES Organization - Dictionary Methodology Committee, July 15, 1991.

[ISO1] "OSI Conformance Testing Methodology and Framework; Part 1: General Concepts," ISO Draft International Standard 9646-1, 1988.

[ISO11] Part 11 ISO TC184/SC4/WG5, "Information Modeling Language EXPRESS: Language Reference Manual," Committee Draft N14, Spiby, P., ed., April 29, 1991.

[ISO31] Part 31 ISO TC184/SC4/WG6, "Conformance Testing Methodology and Framework: General Concepts," Committee Draft, November 15, 1990.

[ISO41] Part 41 ISO TC184/SC4/WG4, "Exchange of Product Data -- Integrated Resources: Fundamentals of Product Description and Support," Committee Draft N16, McKay, A., ed., June 7, 1991.

[ITI91] "Requirements and Recommendations for STEP Conformance Testing," Industrial Technology Institute, internal report, July 1991. {publication due in October 1991 by NIST}

[Jur89] Jurrens, K., "Application of CAD/CAM Data Exchange Through Initial Graphics Exchange Specification (IGES) Flavor Translation, Department of Energy, Contract Number DE-AC04-76-DP00613, September 1989.

[Koh90] Kohout, R., "STEP Data Translations," Student Symposium 1990, Factory Automation Systems Division, National Institute of Standards and Technology, August 1990.

[Mey79] Myers, G., The Art of Software Testing, John Wiley & Sons, New York, NY, 1979.

[Mit90] Mitchell, M., "Development Plan: Validation Testing System," NISTIR 4417, National Institute of Standards and Technology, September 1990.

[NBS88] "Initial Graphics Exchange Specification (IGES), Version 4.0, National Bureau of Standards {now National Institute of Standards and Technology}, NBSIR 88-3813, June, 1988.

[Nij89] Nijssen, G., and Halpin, T., Conceptual Schema and Relational Database Design: A Fact Oriented Approach, Prentice Hall, Englewood Cliffs, NJ, 1989.

[NIST] "Conformance Testing Policy and Procedures," Federal Information Processing Standard, National Institute of Standards and Technology, DRAFT, 1988.

[PDE1] Testing Criteria Requirements Analysis (TC-RAP) Final Report, PDES, Inc. report PMG008.00.00, SCRA, Charleston, SC, June 1989.

[PDE2] Test Report for Context-Driven Integrated Model (CDIM) Application A1, Skeels, J. ed., PDES, Inc. report PTV012.01.00, SCRA, Charleston, SC, April, 1990.

[Sij89] Sijelmassi, R., "An Object-oriented Model for Estelle and its Smalltalk Implementation," National Institute of Standards and Technology, Technical Report NCSL/SNA-89/7, 1989.

[WG5-N19] Wilson, P., "EXPRESS-I Language Reference Manual", ISO TC184/SC4/WG4 Working Draft N19, July, 1991.

[WG4-N14] Part 203 ISO TC184/SC4/WG4, "Configuration Controlled Design," Working Draft N14, Gilbert, M., ed., June, 1991.

[WG4-N15] Palmer, M., Gilbert, M. and Anderson, J., "Guidelines for the Development and Approval of STEP Application Protocols," Version .8, ISO TC184/SC4/WG4 Working Draft N15, May 1991.

[WG5-N5] Danner, W.F. and Yang, Y., "Application Protocol Integration Strategy," ISO TC184/SC4/WG5 Working Draft N5, March 1991.

[WG5-N15] Danner, W.F. and Yang, Y., "AIM Development Methods," ISO TC184/SC4/WG5 Working Draft N15, May 1991.

[WG5-N26] Boyle, A. and Williams, A., "EXPRESS Usage Guide," ISO TC184/SC4/WG5 Working Draft N15, May 1991.

[WG6-N15] Part 33 ISO TC184/SC4/WG6, "Conformance Testing Methodology and Framework: Structure and Development of Abstract Test Suites," Working Draft N15, Owen, J., ed., April 17, 1991.

VIII. Acknowledgements

The author wishes to thank Cita Furlani, Leader of the Product Data Engineering Group, for her continued support and contributions in producing this document.

The basis for the section on AP development came from "Guidelines for the Development and Approval of STEP Application Protocols" by Mark Palmer, Mitchell Gilbert, and Jody Anderson [WG4-N15].

Special appreciation is extended to Larry McKee and the PDES, Inc. Part 203 development team. Their work provided the basis for many of the examples in this document. (See Appendix B for more information.)

The work described in this document was funded by the U.S. Government's Department of Defense Computer-aided Acquisition and Logistic Support (CALS) program. It is not subject to copyright.

Appendix A

STEP Application Protocols

An understanding of the requirements for the development of STEP Application Protocols is a prerequisite for understanding the methodology proposed in this paper. This Appendix will describe the STEP specifications (and projects) which govern APs and testing. The proposed AP development and approval process will be described, along with a series of examples from Part 203, the AP for Configuration Controlled Design [WG4-N14].

Relationship between Application Protocols and STEP

STEP consists of a set of information models that describe product data to be used by multiple industries and application systems. STEP will be a series of standards, called Parts, that support the computer sensible representation and exchange of product data. STEP is being produced in the international standards organization, ISO TC184/SC4[1]. Figure A-1 shows the set of specifications that are the Parts of STEP. These Parts will be referenced throughout this Appendix.

Application Protocols are one category (2xx) of STEP Parts. Each is defined for the purpose of allowing STEP to meet a specific industrial need. Before an AP project is approved, ISO requires evidence that the AP fulfills an industry need. This evidence is to be presented in the form of some conclusive trade assessment such as sponsorship by a recognized trade association, the existence of large national or international program, or an identifiable industry market segment.

While STEP is intended to support a wide variety of applications and industries, each AP selects the elements from STEP that are the most appropriate given the application requirements. The use of these elements is then restricted where necessary to enforce the rules of the application. For example, a cartesian point contains three coordinate attributes. A two dimensional drafting application might not allow the third coordinate to have data associated with it. The parts of STEP that make it capable of supporting diverse applications are the Integrated Resources, e.g. collections of information models that act as libraries for AP specifications.

---

[1] TC184 is the Industrial Automated System Technical Committee and SC4 is the Industrial Data and Global Manufacturing Language Subcommittee

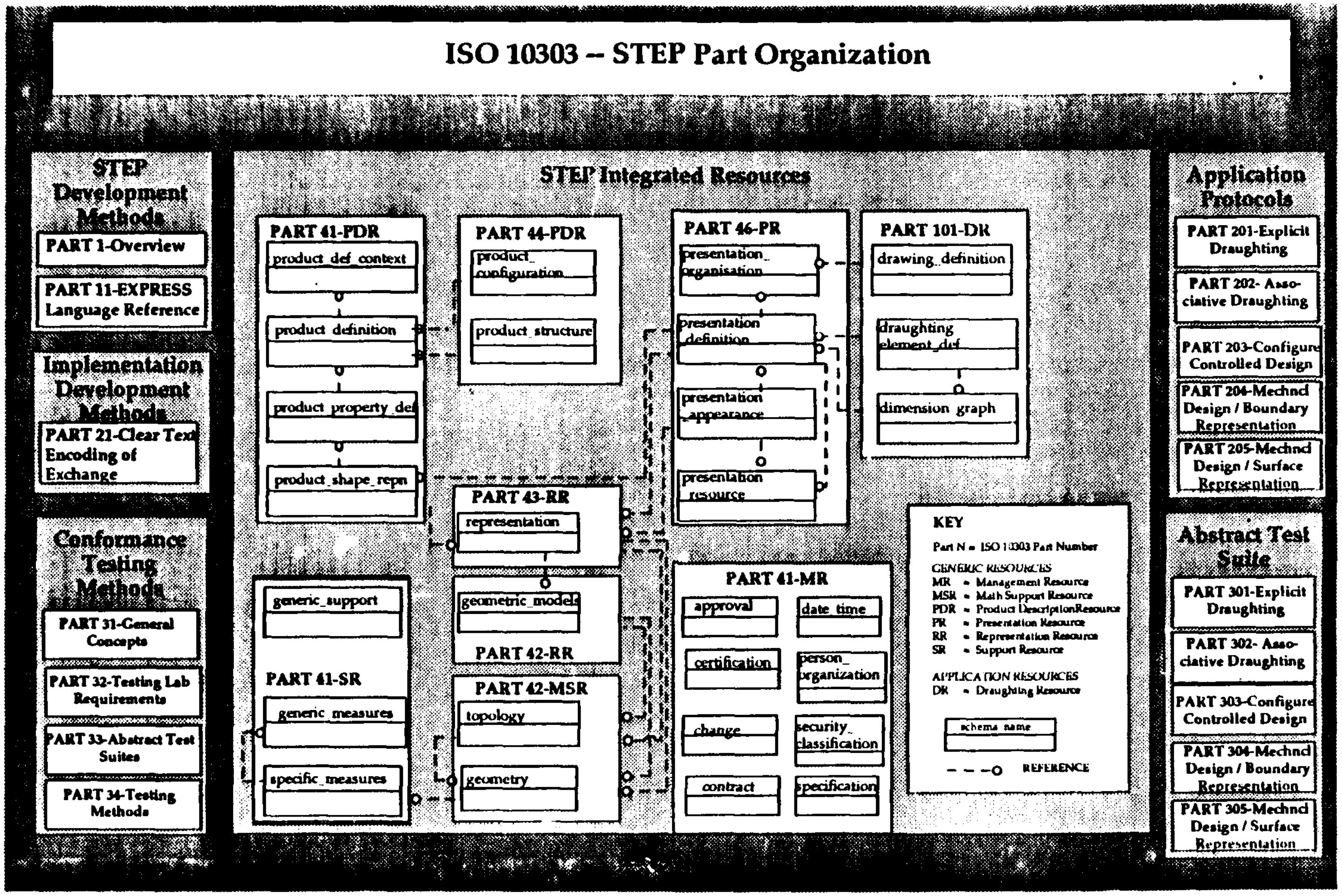


Figure A-1 Current STEP document architecture

WD/MM NIST 8/91

APs are designed to permit practical implementations of STEP. Vendors of computer-aided systems for design, engineering manufacturing and support will be required to comply with specific APs as specified by purchasers of such systems.

The AP Development and Approval Process

This description of the AP development process is summarized from the ISO document "Guidelines for the Development and Approval of STEP Application Protocols" [WG4-N15][2]. In addition, the conformance testing requirements for APs are described in "Conformance Testing Methodology and Framework: Structure and Development of Abstract Test Suites" [WG6-N16].

The following terminology is used in describing the major components of an Application Protocol:

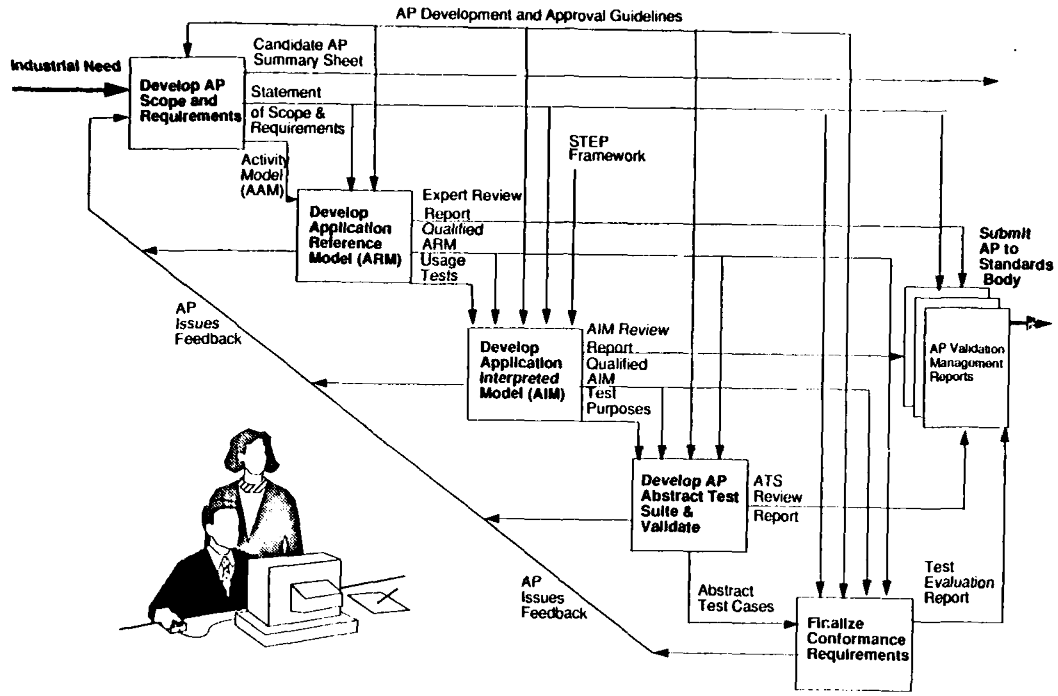
- Scope - a definition of what is included in the AP along with an Application Activity Model (AAM) to describe the processes that are in the AP,
- Application Reference Model (ARM) - a definition of the information requirements in terms familiar to an expert from the appropriate application domain,
- Application Interpreted Model (AIM) - a specification of standardized STEP structures that achieves the requirements described in the ARM,
- Test Purposes - a set of test objectives which are used to determine if the AP achieves the requirements defined in the ARM and the AIM, and
- Conformance Clause - a statement which defines explicitly how complete an implementation must be to be considered conforming and if any options exist in the implementation.

In addition, an optional but important component is:

- Validation Report - a statement which describes the results of the verification and validation activities which have been performed on the AP.

---

[2] The guidelines for AP development provide directives to STEP participants but are not part of the proposed series of standards.



MJM 7/91

Figure A-2 Application Protocol development process

The five phases of AP development (Figure A-2) are described in the AP Guidelines [WG4-N15]:

- Develop AP Scope and Requirements,
- Develop Application Reference Model (ARM),
- Develop Application Interpreted Model (AIM),
- Develop Abstract Test Suite & Validate, and
- Finalize Conformance Requirements.

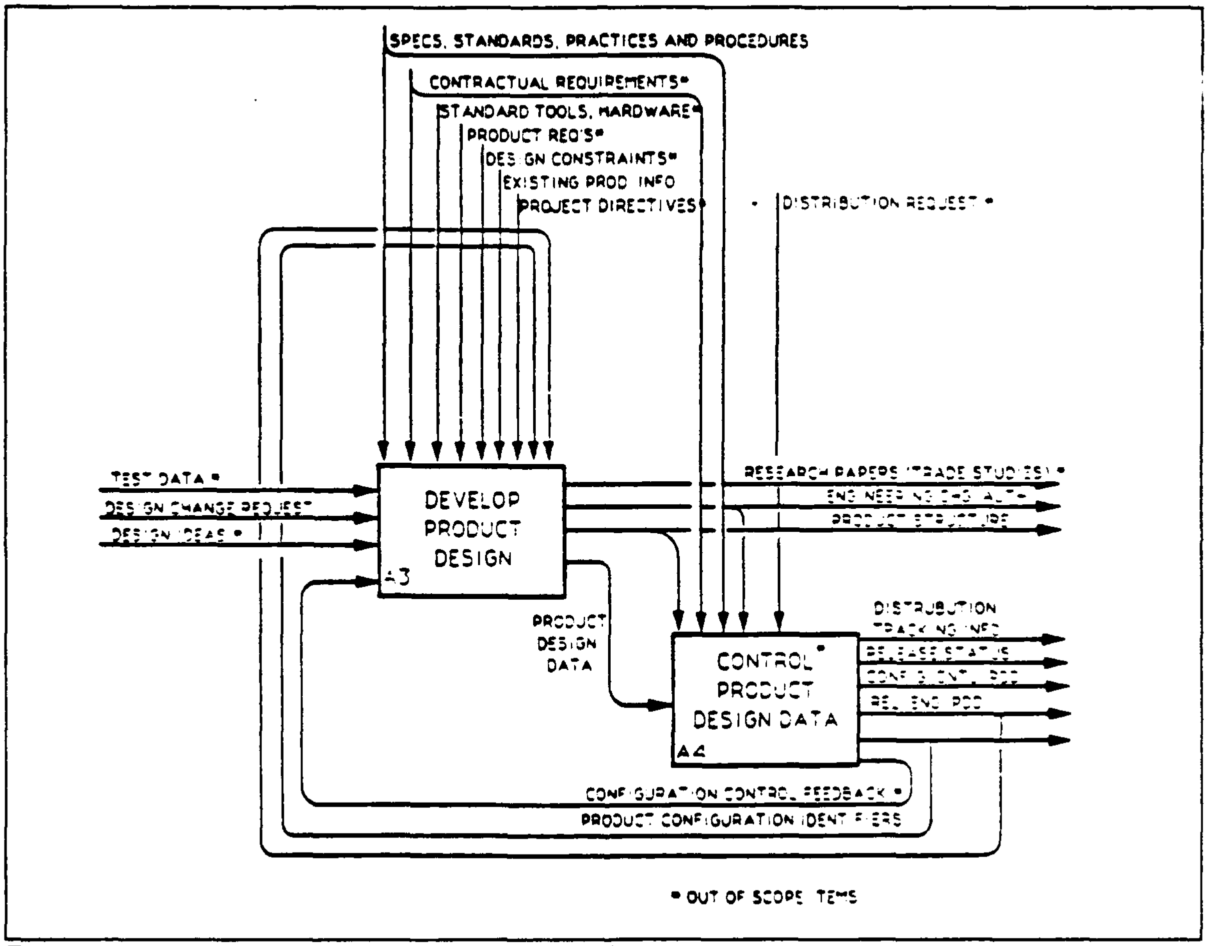


Figure A-3 Application Activity Model (AAM)

In the first phase of AP development the scope, context, and requirements of the AP are defined. A concise statement of scope is formulated to describe 1) the operations to be performed by an application system and 2) how product data will be used to perform these operations. Scope definition is refined using a process modeling technique such as IDEF0 [ICA82]. For each activity of the application the inputs, controls, and outputs are defined. These results are then examined individually to determine if each result is 'in' or 'out' of scope for the AP. During this analysis, example parts and usage scenarios from the application domain are documented. The result of this development effort is documented as an Application Activity Model (AAM) (Figure A-3). The overall application requirements become the evaluation criteria for subsequent steps. Application experts approve these initial decisions.

The second phase is the development of an Application Reference Model (ARM) (Figure A-4). The ARM is a data model that documents the information needs of the application. The ARM is developed using an accepted information modeling technique, e.g., IDEF-1X [ICA85], NIAM [Nij89], or Express and Express-G [ISO11].

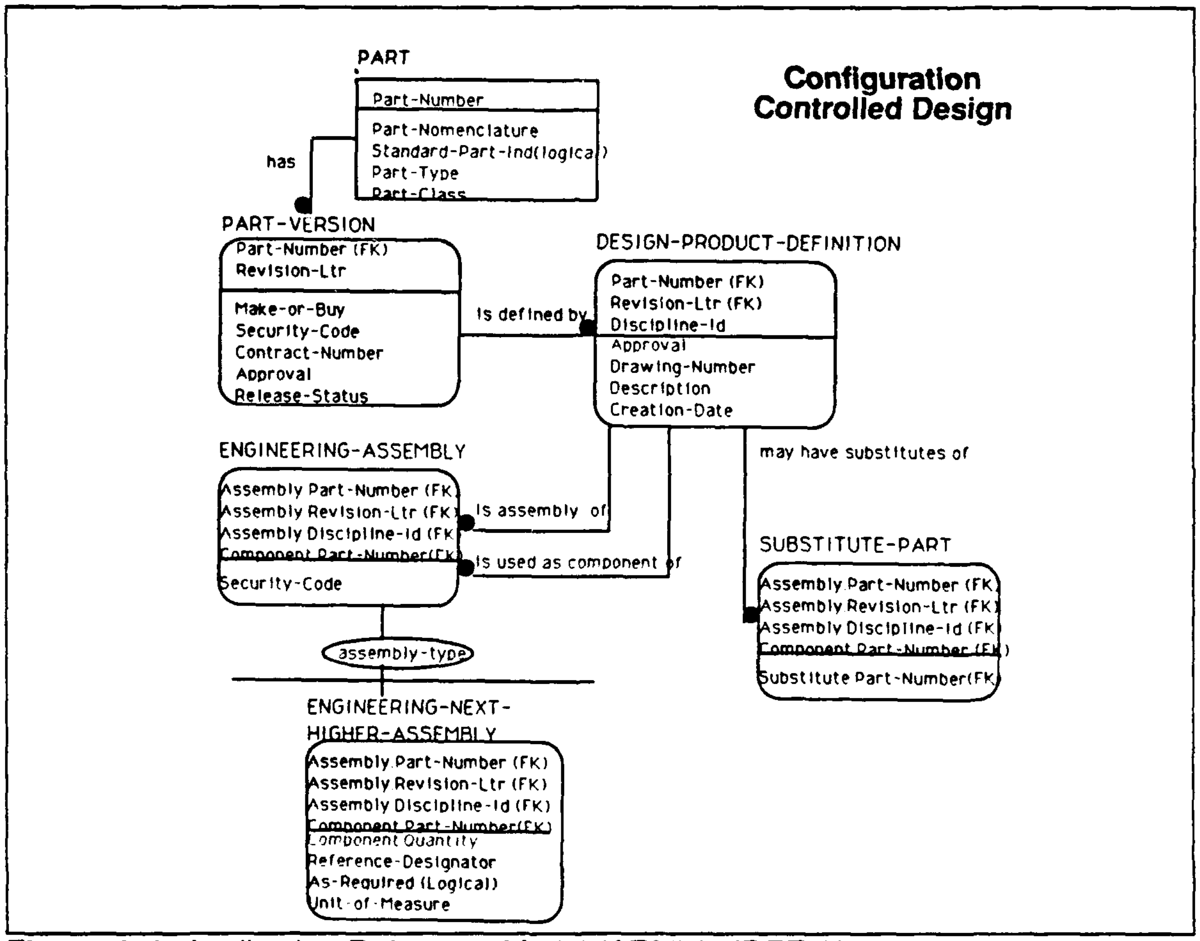


Figure A-4 Application Reference Model (ARM) in IDEF1X

Each application information requirement that is in scope must be defined in the ARM and each element of the ARM must satisfy a documented need of the application. The ARM includes all data elements used, and organizes them into entity definitions.

In the ARM, all definitions and model constructs are described in terms familiar to an expert in the application area.

The AP is organized into manageable constructs by defining Units of Functionality (UOF) within the ARM. A UOF is a construct that performs one specific function according to the accepted practices of the application. For example, an application that requires wireframe geometry would have an UOF entity in its ARM called "wireframe_geometry". The ARM is organized by UOFs to improve its readability to aid the experts in the application area.

The ARM undergoes peer review by a team of application experts to ensure that it satisfies the stated scope and that the ARM is self-consistent. Data from sample products is also used to validate the ARM. The completeness and correctness of the ARM's documented information requirements are evaluated.

The third phase is the development of an Application Interpreted Model (AIM). The fundamental concepts of STEP are enforced on the application via the AIM. The AIM is an Express schema that specifies the formal interpretation of the STEP Integrated Resources to satisfy the information requirements as stated in the ARM. The most appropriate STEP entity for representing a concept depicted in the ARM is selected for use in the AIM specification. The options for using the entities are restricted so that only one method is available for transferring each element of information from the ARM. Without this restriction data exchange could be ambiguous, which could lead to IGES-like 'flavoring'. The AIM may contain additional application-specific constraints to fully represent the functionality of the ARM.

During the AIM interpretation process, the relationship between the constructs in the ARM and in the AIM are documented (Figure A-5). This cross reference establishes correspondence between the functionality of the ARM and the constructs in the AIM. One section of the AIM documentation includes the STEP Integrated Resources that were used. Thus implementors and users of the specification have a self-contained document (Figure A-6). For further discussion see [WG5-N15].

The AIM is now ready to be evaluated for its ability to carry all of the information requirements specified by the ARM. The testing methodology in Section III describes how this may be accomplished. A successful compilation of the Express model ensures that the language syntax is correct. A quality review is done by the AP integration project within ISO TC184/SC4/WG4 to ensure that the style and usage of Express is consistent with the guidelines established for STEP [WG5-N5]. Finally, a joint review with both application experts and STEP experts is conducted to ensure that the ARM information requirements are satisfied by the AIM. As stated in the "Guidelines for the Development and Approval of STEP Application Protocols" [WG4-N15] there is to be no loss or change of meaning in the translation from the ARM to the AIM.

ARM ENTITY/attribute	STEP Part	AIM ENTITY/attribute
-- ASSEMBLY COMPONENT STRUCTURE		(UOF #2)
PART	203	PART
/part_number	41	PRODUCT/id
/part_nomenclature	41	PRODUCT/name
/standard_part_indicator	203	PART/standard_part_indicator
/part_type	41	PRODUCT_CATEGORY
/part_class	203	PART/class -->
	203	part_class
PART_VERSION	41	PRODUCT_VERSION
/revision_letter	41	PRODUCT_VERSION/id
/make_or_buy_code	41	PRODUCT_VERSION/make_or_buy
/security_code	41	SECURITY_CLASSIFICATION
	203	AP203_SECURITY_CLASSIFICATION/ security_level -->
	41	SECURITY_CLASS_LEVEL/level
	203	AP203_SECURITY_CLASSIFICATION/items -->
	203	classified_item = "PRODUCT_VERSION"
/contract_number	41	CONTRACT
	203	AP203_CONTRACT/name

Figure A-5 ARM to AIM cross reference map

In the fourth phase, an abstract test suite for the AP is developed. Each test that will be needed to evaluate the application model is identified. These are called test purposes. Next, the test purposes are fully developed into specific tests called abstract test cases. The abstract test case (ATC) includes a specification of what information is used, how it is used, and the expected outcome of the test [WG6-N15]. Representative product data is used to specify the test. At least one ATC is specified for each test purpose. From the perspective of an ATC, any application system is a black box. The ATC is used to evaluate if the representation of the AIM is accurately represented by an application system and is sufficient to support the exchange of product data, either as an 'input to' or an 'output from.' The abstract test suite is comprised of all of the abstract test cases needed to satisfy the test purposes. The abstract test suite will be a companion standard to the AP. For APs whose scope includes exchange, a STEP physical file which contains all test data used by the test cases is part of the abstract test suite.

The AP guidelines project in ISO has recommended that the abstract test suite be built in conjunction with the development of a prototype implementation. This step is focused on evaluating the AP specification itself. The intent is to prove that the AP is implementable and useful. Currently, AP projects are only required to produce the test purposes and a conformance clause (Figure A-7). However, the abstract test suite is required for conformance testing of the AP [ISO31].

PART

A part is a mechanical piece part or assembly in the context of this Application Protocol. It can be any one of the following classes: assembly, cast, coined, drawn, extruded, forged, formed, machined, molded, rolled, sheared, or sheet_ metal.

EXPRESS Specification:

```
*)
ENTITY part
  SUBTYPE OF (product);
  class: part_class;
  standard_part_indicator: BOOLEAN;
END_ENTITY;
(*
```

Attribute Definitions:

class: The class is the type of part and can indicate the method of creation or material used.
standard_part_indicator: This flag identifies the part as a standard part to either the industry or particular enterprise.

Figure A-6 Fully documented AIM in EXPRESS

Conformance requirements

1. Only the following entities may exist independently:
   -- part
   -- product_model
2. Only those entities in the expanded form of the AIM shall appear.
3. All constraints, as defined by EXPRESS where clauses, unique clauses, and rules, shall always be satisfied.

Figure A-7 Example conformance clause

The last phase is an evaluation of the conformance requirements and the abstract test suite. Once the test purposes and corresponding abstract test suites have been developed, verification of the AP test purposes and conformance clause can begin. In addition to evaluating these elements of an AP, this activity also examines implementation concerns. This might include locating ambiguity or redundancy in the test purpose and the other clause of the AP to which it applies or identifying rules that could not be implemented. The existence of allowable options in an implementation is strongly discouraged. Any implementation options in the conformance clause would be reviewed to see if they could be eliminated. The conformance clause is also judged to determine if it is reasonable and realistic. This step will analyze the completeness of coverage, correctness, and consistency of the abstract test cases with the ARM and AIM.

An evaluation of the AP through simulation, as described in this report, incorporates the use of realistic product data and software to execute a series of test cases. This exercise is needed to ensure that even prototype application systems can be implemented to comply with all aspects of the AP specification. The results of the validation testing are used to identify any needed refinements to the AP.

The development and validation of a STEP AP is an iterative process of progressive detailing and refinement. Each step in this process provides critical feedback to the prior step. Each step requires verification and, where there is sufficient detail, validation. The verification and validation testing uncover defects that must be resolved before the step is complete. Upon completion of the last step, the AP is ready for standardization and implementation by vendors.

The proposed validation testing methodology of this report focuses on the final two phases of AP development. A methodology for implementing the validation testing of APs through a simulation technique and for developing the abstract test suite of an AP are proposed.

Appendix B

PDES, Inc.'s Contribution to Validation Testing

In 1989, PDES, Inc.[1] developed a methodology for evaluating STEP in its Testing Criteria Requirements Analysis Project (TC-RAP) [PDE1]. This methodology was applied in subsequent projects within PDES, Inc. The results of one of these projects is documented in [PDE2]. In these efforts, PDES, Inc. developed and tested small subsets of STEP that provided support for the information needs of applications within the member companies. The scope of these small subsets, called Context-Driven Integrated Models (CDIMs), was defined in such a way that each one could be described and tested in six months.

The methodology included the development of a planning data model to describe, at a coarse level, the information requirements specific to the application context. The boundaries of the application are defined by a process model. Application experts from member companies participated in defining the testing criteria that were used to evaluate the CDIM. The planning model was used to guide the selection of subsets from draft STEP specifications (see Section I). Once the integrated model was completely documented, the CDIM was then released for testing.

Testing of the CDIM was performed by constructing executable tests from test criteria and executing these tests on a test system. Test results were then analyzed to determine the validity of the subset for use by the application. This testing has produced numerous issues and proposed enhancements to the draft STEP specifications.

Experience from PDES, Inc. testing activities found that the majority of the technical issues against an application model were uncovered while attempting to associate realistic data with the application model. A smaller proportion of technical issues were found while attempting to construct the process outputs, even though these tests tended to be more complex, i.e., more conditions had to be satisfied for the test to be successful. This can probably be attributed to two facts: 1) there were fewer issues left to uncover, and 2) the team effort allocated to testing was exhausted before these complex test purposes had been thoroughly tested. However, this type of test assured domain experts that the application model was useful. There was a good correlation between different company requirements for the process output information content, though preferences differed on how to present the information.

---

[1] PDES, Inc. is a consortium of more than 20 member companies and includes NIST as a government associate member. The consortium was formed in 1988 to accelerate the development of STEP and the implementation of Product Data Exchange using STEP (PDES).

Another result was that there were relatively small proportions of the application model that were required frequently. These portions of the application model were of critical importance to usability. This fact can be used to prioritize validation testing efforts.

This work contributed significantly to the progress on AP development methods within the STEP community. The experience gained from NIST participation in the TC-RAP and CDIM development projects provides the basis for the testing methodology proposed in this paper.

NIST-114A (REV. 3-90)	U.S. DEPARTMENT OF COMMERCE NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY	1. PUBLICATION OR REPORT NUMBER NISTIR 4684
		2. PERFORMING ORGANIZATION REPORT NUMBER
	BIBLIOGRAPHIC DATA SHEET	3. PUBLICATION DATE SEPTEMBER 1991

4. TITLE AND SUBTITLE

A Proposed Testing Methodology for STEP Application Protocol Validation

5. AUTHOR(S)

Mary J. Mitchell

6. PERFORMING ORGANIZATION (IF JOINT OR OTHER THAN NIST, SEE INSTRUCTIONS)	7. CONTRACT/GRANT NUMBER
U.S. DEPARTMENT OF COMMERCE NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY GAITHERSBURG, MD 20899	8. TYPE OF REPORT AND PERIOD COVERED

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (STREET, CITY, STATE, ZIP)

Office of the Secretary of Defense
CALS Program Office
Pentagon
Washington, DC 20301-8000

10. SUPPLEMENTARY NOTES

11. ABSTRACT (A 200-WORD OR LESS FACTUAL SUMMARY OF MOST SIGNIFICANT INFORMATION. IF DOCUMENT INCLUDES A SIGNIFICANT BIBLIOGRAPHY OR LITERATURE SURVEY, MENTION IT HERE.)

An Application Protocol (AP) is a specification for a subset of the data described by the Standard for the Exchange of Product Model Data (STEP). Application Protocols are designed to permit practical implementations of STEP. Validation is needed to ensure that the technical solutions provided by the AP will work in a practical sense. This document proposes that the STEP development policy be strengthened to require that Application Protocols be validated prior to being submitted for standardization. Justification for this additional requirement on Application Protocols is provided. The body of the paper describes a series of validation techniques that are appropriate for the development methods used by STEP. A process is proposed under which these validation techniques should be applied. In addition, this paper describes the contribution that AP validation could make to conformance testing.

12. KEY WORDS (6 TO 12 ENTRIES; ALPHABETICAL ORDER; CAPITALIZE ONLY PROPER NAMES; AND SEPARATE KEY WORDS BY SEMICOLONS)

AP; Application Protocols; conformance testing; Standard for Exchange of Product Model Data; STEP; testing; validation

13. AVAILABILITY		14. NUMBER OF PRINTED PAGES
X	UNLIMITED	
	FOR OFFICIAL DISTRIBUTION. DO NOT RELEASE TO NATIONAL TECHNICAL INFORMATION SERVICE (NTIS).	51
	ORDER FROM SUPERINTENDENT OF DOCUMENTS, U.S. GOVERNMENT PRINTING OFFICE, WASHINGTON, DC 20402.	15. PRICE A04
X	ORDER FROM NATIONAL TECHNICAL INFORMATION SERVICE (NTIS), SPRINGFIELD, VA 22161.	

ELECTRONIC FORM